(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
1 March 2001 (01.03.2001)

PCT

(10) International Publication Number
WO 01/15486 A1

(51) International Patent Classification[7]: H04Q 11/04, H04L 29/06

(21) International Application Number: PCT/US00/22853

(22) International Filing Date: 21 August 2000 (21.08.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/150,271 23 August 1999 (23.08.1999) US
09/618,507 18 July 2000 (18.07.2000) US

(71) Applicant *(for all designated States except US)*: NORTEL NETWORKS LIMITED [CA/CA]; World Trade Center of Montreal, 8th Floor, 380 St. Antoine Street West, Montreal, Quebec H2Y 3Y4 (CA).

(71) Applicants and
(72) Inventors: SCOGGINS, Shwu-Yan, Chang [US/US]; 209 Youngford Court, Cary, NC 27513 (US). BROWN, Charles, Michael [US/US]; 1104 Hemingway Drive, Raleigh, NC 27609 (US). JARZEMSKY, David, John [US/US]; 1317 Kintyne Circle, Raleigh, NC 27612 (US). JOYNER, Stanley, Wayne [US/US]; 2021 N. Milpitas Boulevard, Milpitas, CA 95035 (US). SCHELLENBERGER, Kathleen, Kelley [US/US]; 102 Dumnonia Court, Cary, NC 27513 (US).

(74) Agents: PHILLIPS, Steven, B. et al.; Moore & Van Allen, PLLC, P.O. Box 3843, Durham, NC 27702-3843 (US).

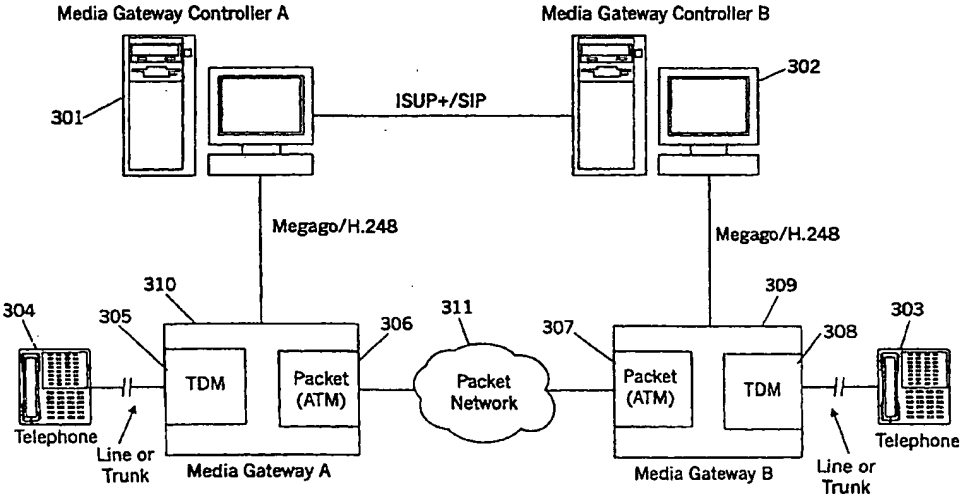
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *With international search report.*
— *Before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments.*

[Continued on next page]

(54) Title: METHOD AND APPARATUS FOR IMPROVED CALL SETUP IN A MULTIMEDIA PACKET NETWORK



(57) Abstract: Events and optional signals are added to the protocol that is used by a media gateway controller (MGC) (301, 302, 401, 402) to control a media gateway (309, 310, 414, 415). These events and signals allow call setup to be monitored by an MGC without actively waiting for call setup, thus avoiding the overhead involved with processing a wait state. The invention can be used to control call processing exclusively, or it can be used in conjunction with transaction pending messages.

WO 01/15486 A1

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

METHOD AND APPARATUS FOR IMPROVED CALL SETUP IN A MULTIMEDIA PACKET NETWORK

DESCRIPTION

Technical Field

This invention is related to multimedia packet networks. Specifically, this invention relates to a mechanism to allow such a packet network to more effectively carry telephony messages, and to more efficiently interface with the public switched telephone network (PSTN).

Background Art

Evolution of the PSTN has accelerated in recent years; however, most of the PSTN still operates on circuit switched, time division multiplexed (TDM) connections. Integrated services digital network (ISDN) bearer channels often provide transport. In parallel with the PSTN, a packet based data network has evolved. This data network has largely been used for Internet traffic and data networking. Although these networks have been mostly separate until recently, the two networks are starting to merge. The merger of these networks requires that voice traffic be carried over packet networks, and further that such packet networks be able to seamlessly

integrate with traditional circuit switched networks, as the two types of networks may carry different call legs of the same call.

FIG. 1 illustrates a typical TDM, PSTN call. Caller 101 places a call to callee 105. The call goes through end office A, 102, over some type of trunk bearer channel to toll office 103, then to end office B, 105, and finally to the callee. Such calls may pass through multiple toll offices, or may be connected directly from one end office to another. In any case, a path of circuits for the call is maintained throughout the call. Signaling between offices is typically provided by an ISUP (ISDN user part) connection. ISUP signaling is well understood and is standard in the telecommunications industry. For more information on ISUP signaling, see the various International Telecommunications Union (ITU) Recommendations pertaining to telephone signaling, including Q.761, Q.764 and Q.931, the most recent versions of which at the time of filing this application are incorporated herein by reference.

FIG. 2 illustrates a call which is similar to the TDM call of FIG. 1; however, in this case, the call is transported from one end office to another (called switch offices, 202 and 204, in this case) via a packet switched network 203. This fact is, in theory, transparent to caller 201 and callee 205. ISUP+ or SIP+ provides signaling in this case. ISUP+ is essentially the same as ISUP except that ISUP+ signals contain extra fields for packet or cell based network information. An International Telecommunications Union

(ITU) recommendation was proposed for ISUP+ as of the filing date of this application as ITU Q.BICC. SIP stands for "session initiation protocol" and is a well-known standard. SIP and SIP+ are described in document RFC 2543, published by the Internet Engineering Task Force (IETF), March, 1999 which is incorporated herein by reference. SIP and SIP+ provide the same type of signaling for control of calls, but are more oriented towards packet based networks.

The network of FIG. 2 has been conceptualized for some time, and standards groups and conference groups have written extensively about how to make such a network work in everyday use. In order for the call leg which is handled by the packet network to seamlessly connect with the call legs handled by TDM switching offices, media provided by one type of network must be converted into media provided by the other. This conversion is referred to as circuit emulation services (CES) in an ATM network. The device that provides this conversion is called a media gateway (MG). In the network of FIG. 2, a media gateway handles each end of the bearer connection through packet network 203. The media gateway terminates bearer media streams from both the switched circuit TDM network, and the packet network. The media gateway and the network it serves may be capable of processing audio and video (hence the term "multimedia packet network"). The media gateway is capable of full duplex media translations. It may also provide other features such as conferencing.

Each media gateway is associated with a media gateway controller (MGC). The media gateway is "dumb" in that it does not have call processing capabilities. The call processing capabilities for the network reside in the MGC. An MGC provides the signaling for call control and controls the call state of a media gateway. The protocol used by the MGC to control the MG is called the media gateway control protocol (or the "Megaco" protocol). Megaco is an application layer protocol which is also described in ITU Recommendation H.248, which shares a common text with the IETF Internet Draft "Megaco Protocol," and which is incorporated herein by reference. The "Megaco Protocol" Internet Draft first became an IETF working document in March, 1999. Within the Megaco protocol, session description protocol (SDP) can be used to describe bearer channel terminations, which are being controlled by the MGC's. SDP is described in document RFC 2327, published by the IETF, April, 1998, which is incorporated herein by reference. Throughout the rest of this disclosure we will refer to Megaco as "Megaco/H.248."

Despite the fact that the theoretical workings of a network like that shown in FIG. 2 have been widely explored, such networks have seen relatively little everyday use. The reason is that there are still problems to be overcome before such networks exhibit the same very high quality of service for voice traffic that users of the PSTN have come to expect. Once such problem stems from the fact that there is no dedicated physical path for

a call through a packet network, and therefore a wait state must be provided so that each node in the network waits for an entire bearer path to be established before timing out.

A packet switched network, used for transport of audio and video media streams, is typically based on asynchronous transfer mode (ATM) frame relay (FR), and Internet protocol (IP) technologies. Public ATM networks generally operate according to the user network interface (UNI). The UNI is described in the book, "ATM User Network Interface (UNI) Specification Version 3.1" by the ATM Forum, published by Prentice Hall PTR, June, 1995, which is incorporated herein by reference. An update to the UNI version 3.1, "ATM User-Network Interface (UNI) Signaling Specification 4.0" was published by the ATM Forum in July, 1996, and is incorporated herein by reference. For private ATM networks, the private network to network interface (PNNI) describes the ATM interface. PNNI is covered in the ATM forum document "PNNI addendum for the network call correlation identifier" published by the ATM forum in July 1999, which is incorporated herein by reference.

In ATM, fixed-length cell carry packetized data. Each cell that is sent through the network has a virtual channel identifier, and other addressing information; however, each node in the network handles many cells that are associated with different media streams. Therefore, there is no provision for a node to control cells for a call throughout a call path. Media gateways and

other nodes in the path must actively wait for call setup throughout the network, consuming processing power and making it difficult to maintain an appropriate level of quality of service. What is needed is a way to more directly monitor the status of a connection, so that valuable processing power is not used to process wait states.

Disclosure of Invention

The present invention solves the above described problem by providing new signals and events to be used in call setup of a connection oriented bearer path so that no wait state is required during the path setup. These signals and events allow processors in the network to switch to other tasks instead of actively waiting for connections to be completed. The signals and events also provide for simpler and more efficient network implementation. The invention can be used exclusively to control call setup. Or it can be used in combination with transaction pending messages.

A media gateway and its MGC at either or both ends of a bearer path can make use of the invention. In describing the invention, we use "originating" and "terminating" to refer to the calling and called ends of the call path, respectively. We use the terms "near-end" and "far-end" to refer to the end of the path relative to where the particular process being discussed is taking place, usually relative to where key messages are being exchanged by a media gateway and an MGC. The terms "near-end" and "far-end" are used

independently of the terms "originating" and "terminating." In connection with call setup, the terms "forward" and "backward" refer to which end initiates the bearer connection through the packet network. "Forward" refers to a process where the originating end sets up the connection, and "backward" refers to a process where the terminating end sets up the connection.

In the Megaco/H.248 protocol, a signal is a request from the MGC to the media gateway to apply some process to network terminations. In the case of TDM terminations, the signals can be in-band signals, such as tones, or out-of-band signals such as those specified by ISUP. In any case, these signals are relayed from the media gateway to the MGC where they are processed. In the case of ATM terminations, most signals are UNI signals or messages. An event is a notification sent from the media gateway to the MGC when a requested state is reached. In the preferred embodiment, the invention is implemented with new events and signals to be used in the Megaco/H.248 protocol. However, the fundamental operation of these events and signals is independent of the underlying bearer network. The signals and events can be mapped to any underlying network such as ATM or frame relay.

According to the invention, a media gateway receives a command from its MGC to add a connection for the bearer path and the command specifies a connection available signal for notifying the MGC if and when the connection is available. We abbreviate "connection available" as "coav."

Coav is both a signal and an event. When an ATM network is used for transport, the command from the MGC triggers the media gateway to send an ATM UNI setup message to the far-end media gateway. The media gateway then initializes the connection with the far-end media gateway. If there is no connection at all for the call, this initialization step involves setup and connect messages being exchanged. In most cases an "add" command is used to direct a media gateway to establish a bearer path, however, some other command, for example, a "move" command can be used. Also, as will be explained in more detail later, there could be a connection which has already been established by the far-end media gateway, in which case, the initialization step simply involves identifying the connection and correlating it with the current call. Once the bearer path has been established, the coav event is sent to the MGC.

If the connection cannot be established, a "connection not available" or "cont" signal is specified. Like coav, cont is both a signal and an event. The MGC can also send cont to the media gateway to disconnect or release the path. The media gateway in this case can respond with the connection not available event. The media gateway will notify the MGC when the connection is disconnected at any time during the call, even if the disconnection occurs because of a network failure. In this case, a notice of failure event may also be sent from the media gateway to the MGC.

As part of the process described above, an optional continuity check can be performed across the bearer path. This check involves one of the media gateways sending a continuity check message to the other media gateway. The media gateway receiving the continuity check then responds with a continuity response. In this way, a bearer path can be confirmed at the end of the setup process to ensure the reliability of the call being handled.

The invention is implemented by software in combination with the hardware of the media gateway and media gateway controller. The software which implements many aspects of the present invention can be stored on a media. The media can be magnetic such as diskette, tape or fixed disk, or optical such as a CD-ROM. Additionally, the software can be supplied via a network. A media gateway is essentially a switching system containing switching fabrics, a computing module, network interfaces, and other resources. The network interfaces are implemented by adapters which are connected to switching fabrics to allow access to the system from the networks. Input/output modules or adapters allow software to be loaded and various maintenance functions to be performed. A computing module contains a processor and memory that execute the software and provide the means to control the operation of the media gateway to implement the invention.

The media gateway controller can also be a switching system, but would more typically be a type of workstation containing a bus such a personal computer interconnect (PCI) bus. A workstation that typically implements the invention includes a plurality of input/output devices and adapters for connection to the necessary networks. A system unit includes both hardware (a central processing unit and memory) and software which together provide the means to implement the media gateway controller.

The invention operates in a network in which media gateways act as endpoints to a call leg being carried on a bearer channel through the network. Each media gateway is controlled by and connected to a media gateway controller. An MGC uses the previously mentioned Megaco/H.248 protocol to control its media gateway, and the invention provides an extension to the Megaco/H.248 protocol.

Brief Description of Drawings

FIG. 1 conceptually illustrates a prior-art telephone connection through the public switched telephone network.

FIG. 2 conceptually illustrates a telephone connection similar to that of FIG. 1, except that one call leg goes through a packet switched network.

FIG. 3 is a block diagram of one network in which the present invention is used.

FIG. 4 is a block diagram of a different network in which the present invention is used.

FIG. 5 is a flowchart illustrating the method of the present invention.

FIG. 6 is a flowchart illustrating the method of the present invention in which a transaction pending message is used.

FIG. 7 is an example message flow diagram that illustrates an example of how the present invention is used.

FIG. 8 is another example message flow diagram that illustrates an example of how the present invention is used..

FIG. 9 is another example message flow diagram that illustrates how the present invention is used.

FIG. 10 is another example message flow diagram that illustrates the use of the present invention.

FIG. 11 is another example message flow diagram that illustrates the use of the present invention.

FIG. 12 is another example message flow diagram that illustrates the use of the present invention.

FIG. 13 is a block diagram of a media gateway that implements the present invention.

FIG. 14 is drawing of one implementation of a media gateway controller that is used with the present invention.

FIG. 15 shows an example of a media which stores software that implements the present invention.

Mode(s) for Carrying Out the Invention

FIG. 3 illustrates one architecture in which the present invention can be used. According to FIG. 3, telephone 304 is where a call originates. Telephone 303 is where a call terminates. Telephones 303 and 304 are shown as illustrations only. In reality, they can be directly connected to the media gateways or can be connected through extensive TDM networks. In the latter case, lines going into the media gateways would actually be TDM trunks. Media gateway A, 310, is the originating media gateway and media gateway B, 309 is the terminating media gateway. The media gateways of FIG. 3 convert voice to ATM. We therefore refer to this network architecture as voice and telephony over ATM, or "VTOA" architecture, however, the same architecture can be used for Internet protocol (IP) telephony. Media gateway controller A, 301, controls media gateway A. Media gateway controller B, 302, controls media gateway B. Alternatively both media gateways can be controlled by a single MGC. Media gateway A includes TDM endpoint 305 and packet endpoint 306. Media gateway B includes TDM endpoint 308 and packet endpoint 307. Packet network 311 serves as the transport network through which bearer channels are established to interconnect calls between the two media gateways. This network and the

endpoints to which it is connected can be frame relay, ATM, IP, or some other type of packet network. For illustrative purposes, most of the discussion refers to an ATM network. The media gateway controllers communicate with each other via ISUP+, SIP, or SIP+. It is also possible to use a nonstandard protocol, specific to the manufacturer of the media gateway controllers and media gateways.

Either a media gateway or a media gateway controller can generate the end-to-end call identifier (EECID), as determined by the network designer. The EECID is used to identify a call leg uniquely across the ATM network, regardless of the number of nodes used in completing the network path. The EECID allows the MGC's, the media gateways, and any nodes in the bearer path to identify the call uniquely. Note that media gateway controller A, 301, controls media gateway A, 310, using the Megaco/H.248 protocol, an application layer protocol for media gateway control. Likewise, media gateway controller B, 302, controls media gateway B, 309, using the same Megaco/H.248 protocol. The media gateway or media gateway controller at either end can generate the EECID, regardless of which end is the originating end for the call and which end is the terminating end for the call.

FIG. 4 illustrates a slightly different architecture in which the invention is used. According to FIG. 4, media gateway controller A, 401, controls media gateway A, 415, using the Megaco/H.248 protocol and media gate-

way controller B, 402, controls media gateway B, 414, using the Megaco/H.248 protocol. In FIG. 4, 403 is the originating telephone and 404 is the terminating telephone. ATM network 411 serves as the transport network. Again, any of the media gateway controllers or media gateways can generate an EECID to identify calls being handled by the network. The main difference between the network of FIG. 4 and the network of FIG. 3 is that the network of FIG. 4 supports digital subscriber loop, or DSL. DSL comes in various types such as aDSL, sDSL and hDSL, and so "xDSL" is used to designate DSL in FIG. 4. In this case each media gateway includes a splitter; 405 in the case of media gateway 415 and 410 in the case of media gateway 414. TDM terminations 406 and 408 and ATM endpoints 407 and 409 each reside in their respective media gateways and allow both data and TDM voice to be transported across the ATM network 411. The splitters 405 and 410 split the voice from the data. The data connection from user terminal 412 is completed through splitter 405 to ATM termination 407 in the case of media gateway A. The data connection from user terminal 413 is completed through splitter 410 to ATM termination 409 in the case of media gateway B. Otherwise, the operation of the network in FIG. 4 is essentially the same as the operation of the network of FIG. 3.

Many aspects of the invention are implemented through enhancements to the previously mentioned Megaco/H.248 protocol. The connection model for the protocol describes logical entities, or objects, within the media

gateway that can be controlled by the media gateway controller. The model relies on extractions, primarily terminations and contexts. A termination sources and/or sinks one or more media streams. A context is an association between a collection of terminations.

In general, an "add" command is used to add terminations to contexts. A termination may be moved from one context to another with a "move" command. A termination exists in, at most, one context at a time. A non-packet termination can exist outside of a context. Property values can be set for terminations by including appropriate descriptors as parameters to the various commands in the Megaco/H.248 protocol. A termination in a context may have its value changed by the "modify" command. Other commands that are important to the implementation of the invention will be discussed later.

As previously mentioned, according to one aspect of the invention an end-to-end call identifier (EECID) is associated with a call, and with a bearer path through the packet network, which completes a call leg. When we say the EECID is associated with a call or a path, we mean that all of the nodes and devices involved in maintaining a call leg are aware of which call to which specific Directory Numbers (DN's), or other user address are associated with each packet of information which flows through the relevant part of the network. Depending on the type of underlying networks and/or protocols

the EECID can be carried across the network in various ways. Details of some possible signaling will be discussed later.

It is preferable to include the EECID in the Megaco/H.248 protocol as part of the stream descriptor in addition to the local control descriptor, local descriptor, and remote descriptor. These descriptors are all part of the stream parameter, a known part of the Megaco/H.248 protocol. It is also possible to include the EECID in the Megaco/H.248 protocol as part of a session descriptor protocol (SDP) term. SDP is a well-known protocol, described in the previously cited IETF RFC 2327, which is used to describe packet terminations, such as IP and ATM terminations within the Megaco/H.248 protocol.

In addition to including the EECID in the Megaco/H.248 protocol, it must be included in other protocols and/or data streams that allow the network to communicate. It is especially important to include the EECID in the ATM cell structure used in the ATM transport network, since the media gateways on the ends of the ATM networks form the ends of the bearer channel carrying the part of the call leg which passes through the packet network. Assuming the packet network shown in FIG. 3 and FIG. 4 is an ATM network implemented according to the UNI standard promulgated by the ATM forum, there is a choice of possible places in an ATM cell where the EECID can be placed. The network prefix is a fixed, required part of the cell, used for routing. The EECID could be placed in the ATM user part.

ATM network routing only uses the first thirteen-byte network prefix of the ATM address. The following 7 bytes of the user part can be used to transport the EECID. Another possible place for the EECID is the ATM subaddressing. The subaddressing field usually only has local significance and can be dropped if it is unused. It can be adapted to implement the EECID of the present invention. Most non-UNI 4.0 compliant ATM networks are currently implemented without using a generic information transport/information element (GIT IE) field; however, the GIT IE field will probably be the best place for the EECID as that field becomes more widely used.

The EECID must also be included in ISUP+ messages, if ISUP+ is used between the two gateway media controllers. FIG. 8 shows an ISUP+ message. Field 801 contains the ISUP information and 802 contains the ISUP+ information, which is essentially directed towards packet based networks, and includes an application transport mechanism field. The EECID is preferably included in the transport mechanism field.

If SIP+ is used between the two media gateway controllers, the EECID is carried as a term in the session description protocol (SDP). Syntax for the session description protocol, which includes the EECID, is "c=eecid: (eecid value)" or "a=eecid: (eecid value)." Note that the terms "c=eecid: (eecid value)" or "a=eecid: (eecid value)" need not be used if the stream descriptor is used to specify the EECID in the Megaco/H.248 protocol.

As previously discussed, either an MGC or a media gateway somewhere in the network selects the EECID, that is, determines a value for the EECID, before it can be associated with a call leg. The EECID can be any arbitrary number that is unique so as to allow correlation of the end-to-end network path between the two media gateways. The choice of the value for the EECID has implications for the call flow. In some cases, the value can only be derived by the network, as with the NCCI as discussed above. Preferably, the value of the EECID is not dependent on the underlying network architecture. A simple way to create an EECID is to simply have the device that is determining the EECID, generate a random number. It is also possible to use a number that is already associated with some part of the network.

A possible value to use for the EECID is a session-ID or call-ID. The session-ID is a random number passed from the MGC to the media gateway. The media gateway can then pass the session-ID to the far end media gateway as an EECID with its initial setup message. The session-ID can also be passed through ISUP+ messages. The session-ID would not be able to be used if the media gateway is to generate the EECID. The call-ID is similar to the session-ID. Both are specified to identify a call solely within an MGC or media gateway.

Finally, the most preferable value for the EECID, assuming a numerical value, which is associated with the network, is used, is the ATM sup-

ported backward network connection identifier (BNC-ID). The BNC-ID is four bytes long and is generated by the media gateway. The media gateway sends the BNC-ID to its media gateway controller for forwarding to the far-end. The BNC-ID is included in the setup command between media gateways to correlate the call.

According to our invention, a package with a signal, called a "connection available" (coav) signal, that explicitly requests the establishment of the packet network path, an event, called a "connection available" (coav) event, that explicitly reports the successful completion of the path, and an event, called "connection not available" (cont), that explicitly reports the failure to establish the requested path can be added to any protocol that is used for call control in any packet-based network. Similarly, the package would include a signal, called a "connection not available" (cont) signal, that explicitly requests the release of the packet network path, and an event, called a "connection not available" (cont) event, that explicitly reports the successful release of the path. If the invention is used with the Megaco/H.248 protocol, these signals and events can be used alone, or with an existing provisional response mechanism implemented through the "transaction pending" command. The events and signals that are used with Megaco/H.248 in both of these alternatives are shown in the following table. The signals and events symbolized by the "coav" and the "cont" symbols are new according to the invention. The EECID, previously discussed, is an optional parameter for

the coav and cont events, hence it is denoted with the letter O. If a media gateway fails to release a packet network path, the media gateway sends a "report failure" (of) event to the media gateway controller. The continuity check, continuity response, and report failure are also part of this package, which we call the "packet pipe" event package:

Symbol	Definition	R	S	Parameter(s)
coav	bearer connection available	X	BR	EECID (O)
cont	bearer connection not available	X	BR	EECID (O)
co1	continuity check	X	TO	
co2	continuity response	X	TO	Duration
of	report failure	X		Duration

R specifies that each symbol is part of an event report. BR indicates a brief tone. TO indicates a timeout tone that stops after the amount of time specified by the duration parameter has passed. Note that the co1 and co2 signal/events are shown for illustrative purposes only to demonstrate the optional use of continuity testing in conjunction with the process of setting up a bearer path. These signals are not required to implement the present invention in Megaco/H.248.

The explicit request alternative has desirable characteristics. In particular, the use of explicit signals and events eliminates the need for the media gateway to maintain the state of an add transaction request. The explicit embodiment also reduces the transaction request state monitoring in the MGC, and eliminates the need for the media gateway to potentially send

multiple transaction pending replies. The explicit signals and events also reduce complexity when multiple add commands are used in a single transaction.

When requesting establishment of a network path, an add command is sent to the media gateway, which then explicitly specifies the "connection available" (coav) signal and event. When requesting release of a network path, a subtract command is sent to the media gateway, which then explicitly specifies the "connection not available" (cont) signal and event. The coav signal is sent only to the ATM termination, which is responsible for the origination of the setup sequence of the network path. The transaction, which specifies these requests, is acknowledged on receipt. The media gateway manifests the coav signal for an ATM network. A notify message of the coav or the "of" event is sent from the media gateway to the MGC upon a connection becoming available or upon failure to establish the connection, respectively. Note that the use of the coav and cont signals are optional. If they are not used, the initiation of the establishment or release of the path is implied by the add or subtract command, respectively.

The continuity check and response can be used automatically by the media gateway without an instruction from the MGC. However, these two events/signals can also be requested by the MGC during call processing. An additional characteristic of this approach is that embedded signals and

events can be used to allow for additional processing to be invoked automatically for such things as continuity checking of the network path.

An alternative way of making a connection request is based on the Megaco/H.248 provisional response "transaction pending" reply wait state. This command is used when a command is received but pending for completion of processing. The media gateway can respond to the MGC with a command "transaction pending" response, so that the MGC won't be blocked for the completion response. When the media gateway finishes executing the command, it can then send the "transaction reply" message to acknowledge that the original command has been successfully completed or has failed. In addition to using the coav signal explicitly, the connection request is expressed implicitly by the add command. The rationale behind this approach is that the packet connection does not exist until it is added to the context. Therefore, the add command implies setting up the bearer connection.

The EECID is present in either the stream descriptor or the termination descriptor for the network path. The media gateway must use the EECID to determine if it needs to initiate the network path setup. The media gateway will keep a record of all requests received from other media gateways for setup of a network path. When the add command is received, the media gateway will determine if a bearer path setup request with the specified EECID has been received. If a network path associated with the EECID

exists, then the network path bearer connection already exists and the correlation is reported back to the MGC via a transaction reply. If no pending network path is found with the same EECID, then the path initialization is invoked. In this approach, there won't be a coav signal and bearer connection available event notification. If the media gateway determines that there will be sufficient delay setting up the bearer connection to cause the transaction request to time out, the media gateway will respond to the media gateway controller with a transaction pending response. Upon completing the bearer connection, the media gateway will respond to the MGC with a transaction reply indicating success or failure of the attempt.

The method described immediately above can incur processing overhead in determining whether or not network path setup is required. Another negative consideration is that there is no mechanism to use embedded signals and events to allow for automatic processing of subsequent actions such as continuity checking of the network bearer path. The MGC has to issue a separate message for continuity checking and response. This combination at least eliminates the need for the media gateway to search through pending network path requests to determine if a network path setup is required. If the coav signal is present, the setup will begin immediately. If the two embodiments are used simultaneously, accommodations have to be made to eliminate redundant messaging to report the completion of the add command and the coav or cont event.

Figures 5 and 6 illustrate the overall method of the invention. FIG. 5 is a flowchart that shows the explicit call setup method. At 501 the add command is sent from the MGC to the media gateway. The add command can specify any signal and response that the media gateway will understand. In this case, the signal and response is coav for connection available, and/or cont for connection not available. At 502 the add command is confirmed. In this embodiment, the confirmation takes the form of an immediate transaction reply message to the MGC. At 503 the bearer connection is initialized.

The bearer connection can be initialized either by the media gateway setting up the bearer connection, called a forward setup, or the media gateway simply identifying an existing bearer connection, which has been set up by a far-end media gateway, called a backward setup. In either case, the media gateway waits until the bearer path is completely established at 504, and responds with a coav event at 505. The MGC will typically notify the far-end MGC of this occurrence at 506. An optional continuity check is shown as being performed at 507. This continuity check consists of a continuity check message being sent to the far-end media gateway, and a response being received. If the connection is not available in FIG. 5, a cont event is sent to the MGC at 508, and the far end is notified at 509. The process then ends at 510.

The flowchart of FIG. 6 illustrates the implicit call setup method used with the new signals and events of the invention. At 601 an add command is given by the MGC to the media gateway as before. This time, however, the media gateway responds with a transaction pending message at 602, which forces the MGC to wait until it receives further instructions from the media gateway. At 603, the media gateway again initializes the bearer path. Only after the bearer path is fully established is the transaction reply sent to the MGC at 608. A coav event is returned to the MGC at 609. The far-end is notified at 610. Again, a continuity check is optionally performed at 611.

In FIG. 6, if the bearer path has not been established at 604 after a time-out at 605, the cont event is returned to the MGC at 606. The far end is notified at 607. In this case, a continuity check can still be performed at 611 to further check the status of the network.

To illustrate the detail of the invention, Figures 7-12 present detailed signal flows showing the setup of bearer path connections in a multimedia packet network. There are literally dozens of possible signal flows, which could be implemented to make use of the invention. The signal flows presented here should be considered as examples only. When we refer to implicit versus explicit setup, we are using the terminology discussed above for explicit versus implicit signaling and events. When we refer to forward setup versus backward setup we are referring to which end of the network is performing the bearer path setup relative to the originating end of the net-

work. If the originating end of the network is also setting up the bearer connection we have a forward setup. If the originating end of the network is passing information to the terminating end and the terminating end is setting up the bearer connection, we have a backward setup. In reference to FIG. 9, all messages are discussed. For the other message flow diagrams, only new messages, which are important to illustrating the differences between those examples and previous examples, are discussed. The letters A and B correspond to the ends of the network path as shown in the network diagrams of Figures 3 and 4.

In Figures 7-12, the TDM termination is a logical representation of a TDM line and an ATM termination is a logical representation of an ATM network connection. Although an ATM termination is illustrated in all cases, the invention is not limited to use of an ATM network for the bearer connection. The invention is also applicable with other connection-oriented networks such as frame relay networks.

Turning to FIG. 7, an explicit forward setup is illustrated. At 710, a notify message indicating an offhook condition is sent from media gateway A to MGC A. At 711, MGC A responds with an add command. At 716, media gateway A replies with a transaction reply. At 713 and 701, the two media gateway controllers negotiate connection parameters. At 714, MGC A sends a pipe connect request to MGC B. In this case, at 702, MGC B sends an add command to media gateway B with explicit instructions for setting up

the bearer path with the connection signal coav and the event coav when the connection is available. At 703, media gateway B immediately responds to MGC B with a transaction reply signal; the transaction reply signal is in response to an add command. This transaction reply does not mean that the add command is completed. Rather, the transaction reply simply means that media gateway B is working on adding the ATM termination. Media gateway B chooses an EECID at 702 and sends the EECID back to MGC B at 703. MGC B passes the EECID to MGC A at 707. At 715, MGC A sends the add command with the EECID and explicit event and signal coav. Media gateway A immediately replies to MGC A with a transaction reply at 717. At 704, the UNI setup message is sent from media gateway A through the ATM network to media gateway B. A connect message is sent from media gateway B to media gateway A to indicate the bearer path is accepted at 705. Media gateway B uses the EECID to associate the call and the bearer path. This prevents an unauthorized bearer connection from being set up. Then MGC B notifies media gateway B at 706.

After receiving a UNI connect message from media gateway B, media gateway A notifies MGC A that the coav event has occurred at 708. In the above example, the MGC B cannot add the ATM termination until the UNI service has been set up. This limitation comes about because the EECID is needed to create the ATM termination. At 718, MGC B is notified by MGC A through ISUP+ or other means that the bearer path (packet pipe) has been

established. The process is completed with the pipe connect complete ack message at 712.

FIG. 8 illustrates the signal flows for explicit backward setup. At 801, MGC A chooses the EECID and passes it to media gateway A. There is an embedded continuity check applied after the coav event occurs and there is a return continuity check associated with the continuity check event. At 812, MGC A passes the EECID to MGC B. At 802, media gateway A sends a transaction reply to the add command to acknowledge that the transaction is accepted. Once again, the add command has not been fully executed. At 811, MGC B sends an add command to MG B. This command asks for a bearer path to be set up using signal=coav. At 803, media gateway B sends the ATM UNI setup message with an EECID to media gateway A. An event notification on coav with embedded event co1 and signal co2 is explicitly requested. Upon receipt of continuity check co1 response co2 will be given. This command also asks for a bearer path to be set up using signal=coav. After the connection is set up, media gateway B responds at 805 with a coav event. At 809 MG A also notifies MGC A that event=coav has occurred. At 806, the continuity check signal is applied by media gateway A since the coav has occurred. At 807, media gateway B applies the continuity check response signal since it receives the continuity check event. At 808, media gateway B notifies its media gateway controller that the continuity check event has occurred. Similarly, media gateway A notifies its media gateway

controller that the coav and the continuity check return events have both occurred at 810.

FIG. 9 illustrates the message flows for implicit backward setup. In this case, the EECID is assigned by the media gateway. At 901, media gateway A chooses the EECID and sends a transaction pending response with the EECID to its media gateway controller so that the media gateway controller waits for the setup. At 902, media gateway B passes the EECID in the UNI setup message to media gateway A. At 903, when the connection setup is complete, media gateway A sends a transaction reply to the add command and the EECID to media gateway controller A. Media gateway B immediately sends a transaction reply to the add command to MGC B.

FIG. 10 also illustrates an implicit forward setup. In this case a media gateway controller creates the EECID. Also, the co1 and co2 continuity checking messages are used. At 1001, an add command with the EECID is sent from media gateway controller A to media gateway A. At 1002, media gateway A responds to its MGC with a transaction pending command. Note that media gateway A still uses the EECID for establishing the connection with media gateway B. 1003 and 1004 illustrate the continuity check and continuity check response, respectively.

Figures 11 and 12 illustrate what happens when a failure occurs. In FIG. 11, at 1101, MG B can't accept the UNI setup due to an error. Any

number of things could cause the error. One possibility is that there is no EECID known at MG B that matches the one in the UNI setup message. At 1104, MG B reports failure (of) to MGC B. At 1102 MG A times out or receives a reject message from MG B. MG A reports failure (of) to MGC A at 1103. At 1105, and 1106, MGC a and MGC B exchange messages to disconnect the pipe connection. Messages like that shown at 1105 and 1106 can come from either MGC A or MGC B.

In FIG. 12, the EECID is created and sent from MG A to MGC A at 1201. MG A sends the UNI setup message at 1202, but it is rejected at 1206. After the add command is confirmed at 1205, a report of failure (of) is sent from MG A to MGC A at 1208. A report of failure is also sent from MG B to MGC B at 1207.

FIG. 13 illustrates a conceptual, functional block diagram of a switching system, which can be used to implement a media gateway, which in turn implements the invention. Computing module 1301 includes a central processing unit, memory, and supporting circuitry. This computing module, together with any computer program code stored in the memory, is the means for controlling the overall operation of the switching system to perform the method of the invention. TDM switching fabric 1302 is for switching TDM channels and is controlled by the computing module. Input/output (I/O) module 1304 is also connected to the processor of computing module 1301 and includes media devices to load computer program code as well as

connections for workstations or other equipment for control and maintenance of the switching system. TDM network access module 1303 serves as a TDM network interface and is connected to TDM switching fabric 1302, both of which are managed by the computing module 1301. Circuit emulation system 1305 provides circuit emulation services, converting TDM voice to packets such as ATM cells. Packet switching fabric 1306 sends and receives packets on the packet network through packet network interface 1307.

FIG. 14 illustrates a workstation, which can be used to implement a media gateway controller according to the present invention. I/O devices such as keyboard 1402, mouse 1403 and display 1404 are used to control the system. One or more of these devices may not be present in normal operation. System unit 1401 is connected to all devices and contains memory, media devices, and a central processing unit (CPU) all of which together form the means to implement the present invention. Network interfaces are normally implemented via adapter cards plugged into a bus, however, for the sake of simplicity they are shown graphically as interface 1405.

As previously mentioned, appropriate computer program code in combination with appropriate hardware implements most of the elements of the present invention. This computer program code is often stored on storage media. This media can be a diskette, hard disk, CD-ROM, or tape.

The media can also be a memory storage device or collection of memory storage devices such as read-only memory (ROM) or random access memory (RAM). Additionally, the computer code can be transferred to the workstation over the Internet or some other type of network. FIG. 15 illustrates one example of a media. FIG. 15 shows a diskette of the type where magnetic media 1502 is enclosed in a protective jacket 1501. Magnetic field changes over the surface of the magnetic media 1502 are used to encode the computer program code. In this way the computer program code is stored for later retrieval.

We have described specific embodiments of our invention, which provides an end-to-end call identifier (EECID) to uniquely identify a call leg across a packet network, regardless of the number of nodes used in completing the network path. One of ordinary skill in the networking and computing arts will quickly recognize that the invention has other applications in other environments. In fact, many embodiments and implementations are possible. The following claims are in no way intended to limit the scope of the invention to the specific embodiments described.

We claim:

CLAIMS

1. In a media gateway, a method of establishing a bearer path through a multimedia packet network, the method comprising the steps of:

receiving **(501)** a command from a media gateway controller (MGC) to establish a connection for the bearer path;

confirming **(502)** the command with the MGC;

initializing **(503)** the connection with a far-end media gateway; and

sending **(505)** a connection available event to the MGC if and when the bearer path has been established so that call setup can be completed without processing a wait state.

2. The method of claim 1 wherein the command specifies a connection available signal and a request for notification of connection available and connection not available events.

3. The method of claim 1, further comprising the step of sending **(508)** a connection not available event to the MGC if and when a determination has been made that the bearer path cannot be established.

4. The method of claim 2 further comprising the step of sending (508) a connection not available event to the MGC if and when a determination has been made that the bearer path cannot be established.

5. In a media gateway, a method of releasing a bearer path through a multimedia packet network, the method comprising the steps of:

receiving a command from a media gateway controller (MGC) to release a connection for the bearer path;

confirming the command with the MGC;

releasing the connection with a far-end media gateway; and

sending a connection not available event to the MGC if and when a determination has been made that the bearer path has been released.

6. The method of claim 5 wherein the command specifies a connection not available signal and a request for notification of a connection not available event.

7. In a media gateway controller (MGC), a method of directing a media gateway to establish a bearer path through a multimedia packet network, the method comprising the steps of:

sending **(601)** a command to the media gateway to establish a connection for the bearer path;

confirming the command with the media gateway; and

receiving **(609)** a connection available event from the media gateway if and when the bearer path has been established so that call setup can be completed without processing a wait state.

8. The method of claim 7 wherein the command specifies a connection available signal and a request for notification of connection available and connection not available events.

9. The method of claim 7, further comprising the step of receiving a connection not available event from the media gateway if and when a determination has been made that the bearer path cannot be established.

10. The method of claim 8 further comprising the step of receiving a connection not available event from the media gateway if and when a determination has been made that the bearer path cannot be established.

7

11. In a media gateway controller (MGC), a method of directing a media gateway to release a bearer path through a multimedia packet network, the method comprising the steps of:

sending a command to the media gateway to release a connection for the bearer path;

confirming the command with the media gateway; and

receiving a connection not available event from the media gateway if and when a determination has been made that the bearer path has been released.

12. The method of claim 11 wherein the command specifies a connection not available signal and a request for notification of a connection not available event.

13. A computer program product **(1501, 1502)** for enabling a media gateway to establish a bearer path through a multimedia packet network, the computer program product having a media with a computer program embodied thereon, the computer program comprising:

computer program code for initializing a connection for the bearer path in response to receiving a command from a media gateway controller (MGC) to establish a connection for the bearer path;

computer program code for confirming commands with the MGC;

computer program code for sending a connection available event to the MGC if and when the bearer path has been established so that call setup can be completed without processing a wait state; and

computer program code for sending a connection not available event to the MGC if and when a determination has been made that the bearer path cannot be established.

14. The computer program product of claim 13 wherein the computer program further comprises computer program code for releasing the connection for the bearer path in response to receiving a command from the MGC to release the connection for the bearer path and confirming that the

connection has been released by sending the connection not available event.

15. The computer program product of claim 13 wherein the command to establish a connection specifies a connection available signal and a request for notification of the connection available and connection not available events.

16. The computer program product of claim 14 wherein the command to establish a connection specifies a connection available signal and a request for notification of the connection available and connection not available events and the command to release the connection specifies a connection not available signal and a request for notification of a connection not available event.

17. A computer program product **(1501, 1502)** for enabling a media gateway controller (MGC) to direct a media gateway to establish a bearer path through a multimedia packet network, the computer program product

having a media with a computer program embodied thereon, the computer program comprising:

computer program code for sending a command to the media gateway to establish a connection for the bearer path;

computer program code for confirming commands with the media gateway;

computer program code for receiving a connection available event from the media gateway if and when the bearer path has been established so that call setup can be completed without processing a wait state and for receiving a connection not available event if and when a determination has been made that the bearer path cannot be established.

18. The computer program product of claim 17 wherein the computer program further comprises computer program code for sending a command to the media gateway to release the connection for the bearer path and for receiving the connection not available event in response.

19. The computer program product of claim 17 wherein the command to establish a connection specifies a connection available signal and a

request for notification of the connection available and connection not available events.

20. The computer program product of claim 18 wherein the command to establish a connection specifies a connection available signal and a request for notification of the connection available and connection not available events and the command to release the connection specifies a connection not available signal and a request for notification of a connection not available event.

21. Apparatus for establishing a bearer path through a multimedia packet network, the apparatus comprising:

means for initializing **(1301, 1304, 1306, 1307)** a connection for the bearer path in response to receiving a command from a media gateway controller (MGC) to establish a connection for the bearer path;

means for sending **(1301, 1304)** a connection available event to the MGC if and when the bearer path has been established so that call setup can be completed without processing a wait state; and

means for sending **(1301, 1304)** a connection not available event to the MGC if and when a determination has been made that the bearer path cannot be established.

22. Apparatus for directing a media gateway to establish a bearer path through a multimedia packet network, the apparatus comprising:

means for sending **(1401, 1405)** a command to the media gateway to establish a connection for the bearer path;

means for receiving **(1401, 1405)** a connection available event from the media gateway if and when the bearer path has been established so that call setup can be completed without processing a wait state and for receiving a connection not available event if and when a determination has been made that the bearer path cannot be established.

23. A media gateway for establishing a bearer path through a multimedia packet network under the control of a media gateway controller (MGC), the media gateway comprising:

a switching fabric **(1302, 1306)**;

one or more network interfaces **(1303, 1307)** connected to the switching fabric; and

a computing module **(1301)** connected to the switching fabric for controlling the switching fabric according to a computer program to enable the media gateway to receive commands from a media gateway controller (MGC) to establish and release connections for the bearer path, and send a connection available event to the MGC if and when the bearer path is established and a connection not available event to the MGC if and when a bearer path is not established.

24. The media gateway of claim 23 wherein a command to establish a connection specifies a connection available signal and a request for notification of the connection available and connection not available events and a command to release the connection specifies a connection not available signal and a request for notification of a connection not available event.

25. A programmed computer system **(1401, 1405)** that is operable as a media gateway controller, the programmed computer system having connections **(1405)** for at least one media gateway, the programmed computer system including a computer program comprising:

computer program code for sending a command to the media gateway to establish a connection for a bearer path;

computer program code for confirming commands with the media gateway;

computer program code for receiving a connection available event from the media gateway if and when the bearer path has been established so that call setup can be completed without processing a wait state and for receiving a connection not available event if and when a determination has been made that the bearer path cannot be established.

26. The computer system of claim 25 wherein the computer program further comprises computer program code for sending a command to the media gateway to release the connection for the bearer path and for receiving the connection not available event in response.

27. The computer system of claim 25 wherein the command to establish a connection specifies a connection available signal and a request for notification of the connection available and connection not available events.

28. The computer system of claim 26 wherein the command to establish a connection specifies a connection available signal and a request for notification of the connection available and connection not available events and the command to release the connection specifies a connection not available signal and a request for notification of a connection not available event.

29. A packet network for supplying bearer paths for multimedia calls comprising:

at least one media gateway **(309, 310, 414, 415)** for initializing and releasing bearer path connections according to commands received, the media gateway also operable to send a connection available event if and when the bearer path is established and a connection not available event if and when a bearer path is not established; and

at least one media gateway controller (MGC) **(301, 302, 401, 402)** connected to the media gateway, the MGC operable to send the commands to media gateway and to receive the connection available event and the connection not available event.

30. The network claim 29 wherein the commands include a command to establish a connection that specifies a connection available signal and a request for notification of the connection available and connection not available events.

31. The network of claim 30 wherein the commands further include a command to release the connection that specifies a connection not available signal and a request for notification of a connection not available event.

FIG. 1
PRIOR ART

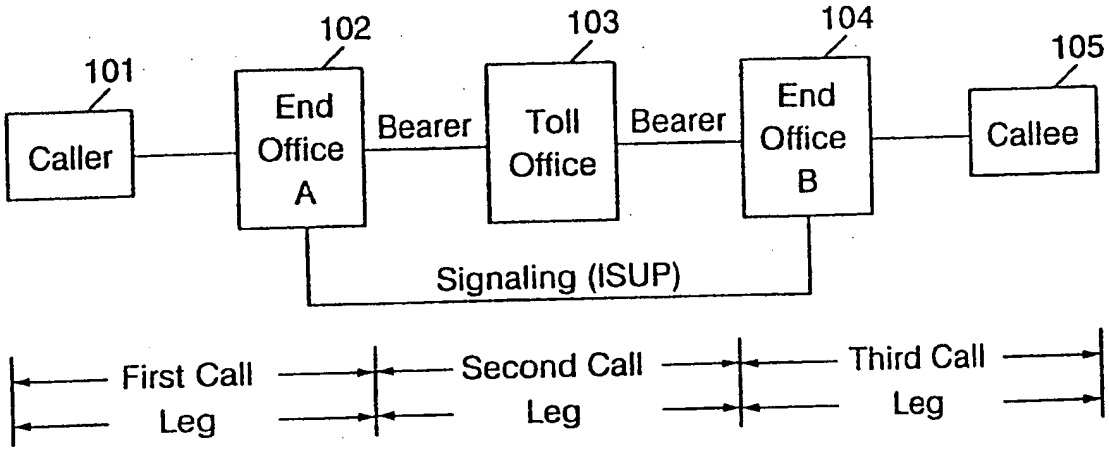


FIG. 2
PRIOR ART

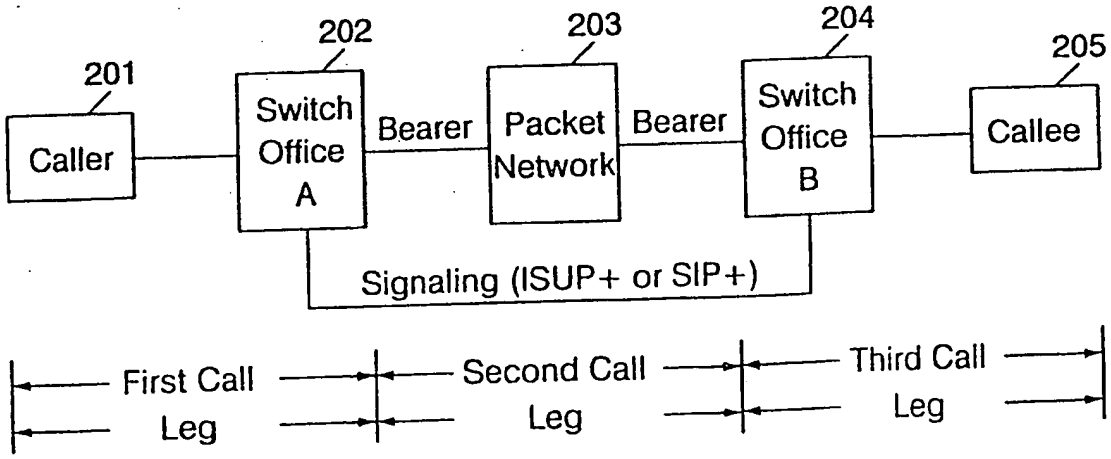


FIG. 3

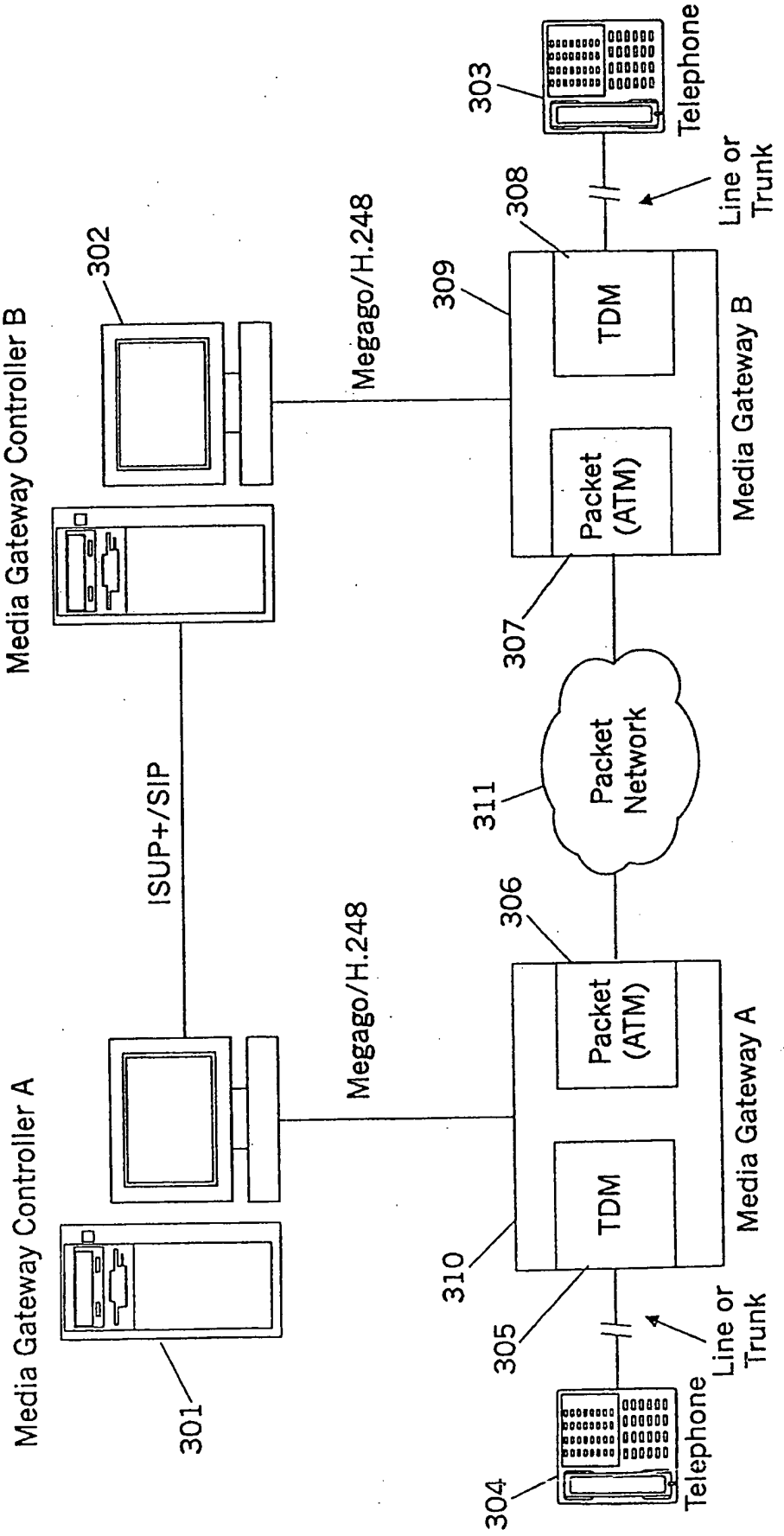

Media Gateway Controller A
Media Gateway Controller B
ISUP+/SIP
Megago/H.248
Megago/H.248
Packet Network
Media Gateway A
Media Gateway B
TDM
Packet (ATM)
TDM
Packet (ATM)
Telephone
Telephone
Line or Trunk
Line or Trunk
301
302
303
304
305
306
307
308
309
310
311

FIG. 4

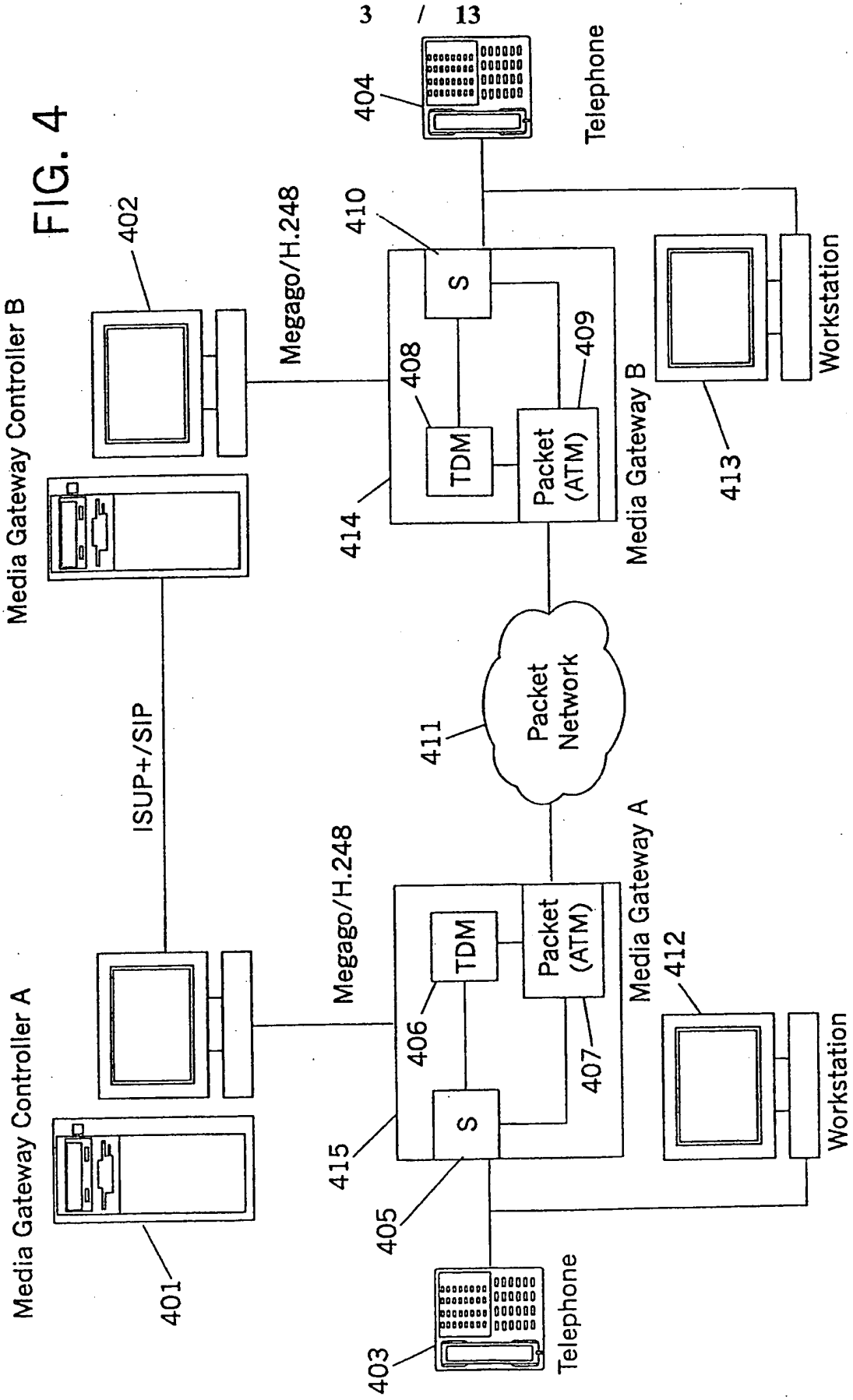

Media Gateway Controller A
Media Gateway Controller B
ISUP+/SIP
Megago/H.248
Megago/H.248
Media Gateway A
Media Gateway B
Packet Network
TDM
Packet (ATM)
TDM
Packet (ATM)
S
S
Telephone
Telephone
Workstation
Workstation
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415

FIG. 5

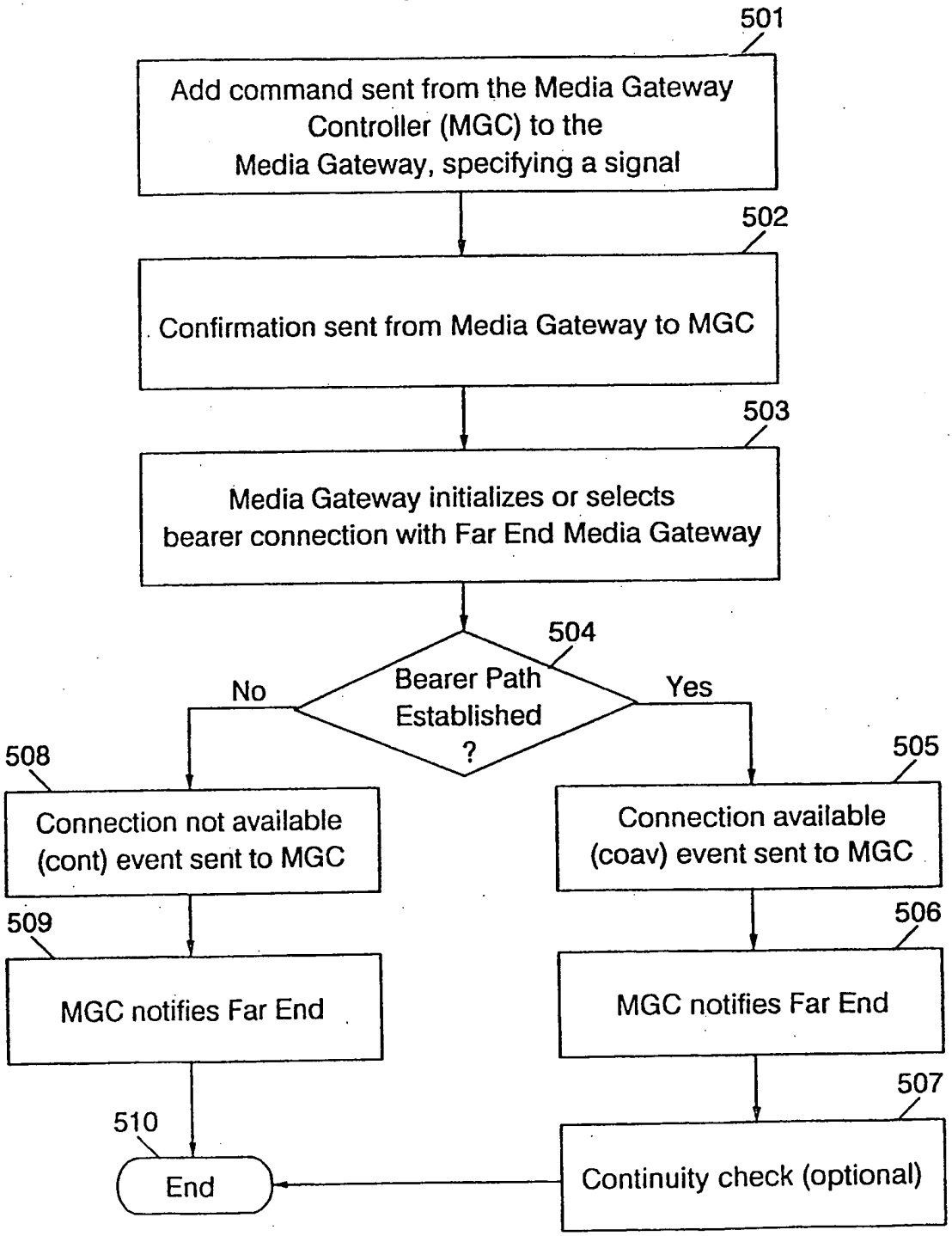

501
Add command sent from the Media Gateway Controller (MGC) to the Media Gateway, specifying a signal
502
Confirmation sent from Media Gateway to MGC
503
Media Gateway initializes or selects bearer connection with Far End Media Gateway
504
Bearer Path Established ?
No
Yes
508
Connection not available (cont) event sent to MGC
505
Connection available (coav) event sent to MGC
509
MGC notifies Far End
506
MGC notifies Far End
507
Continuity check (optional)
510
End

FIG. 6

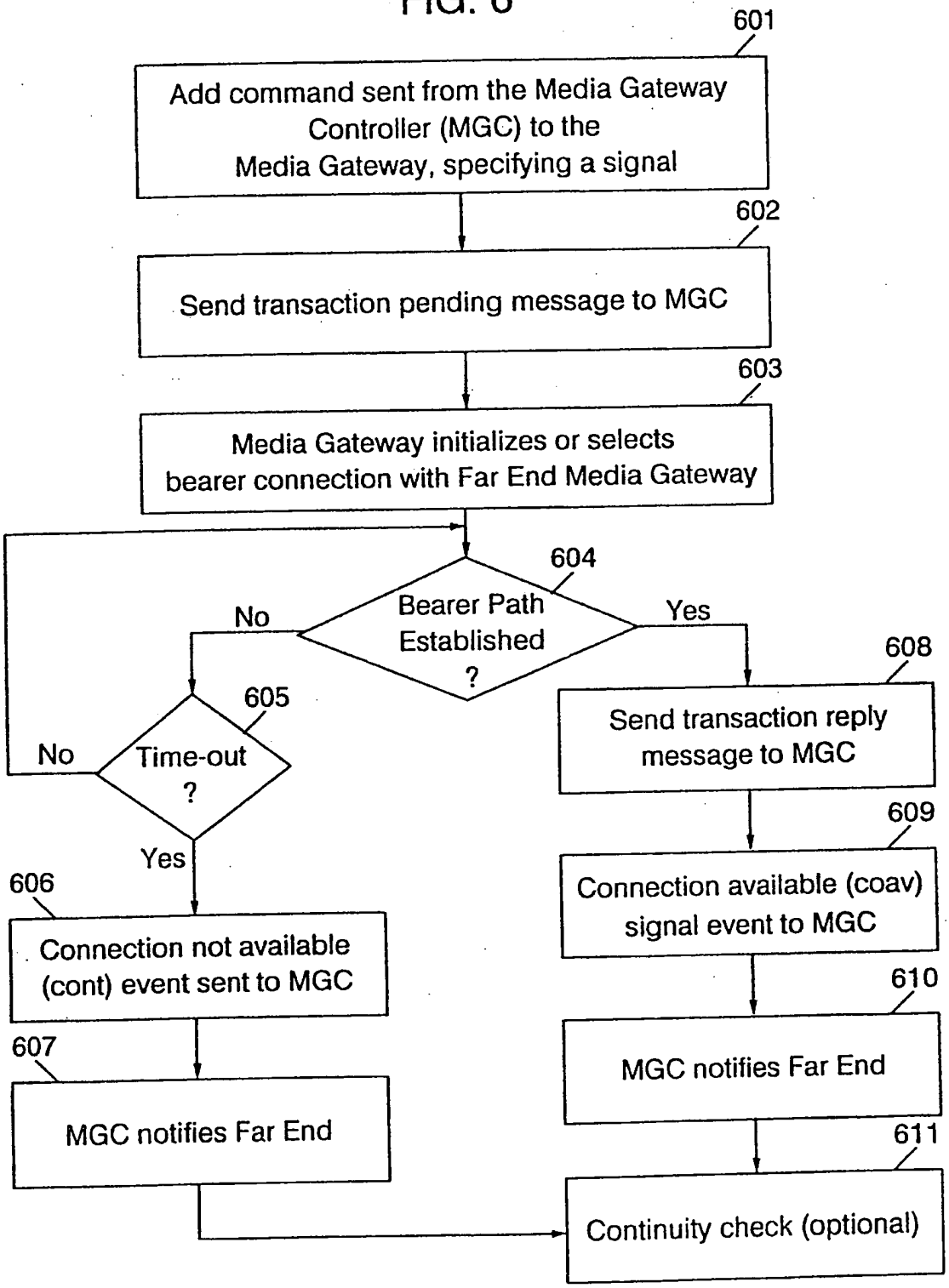

601
Add command sent from the Media Gateway Controller (MGC) to the Media Gateway, specifying a signal
602
Send transaction pending message to MGC
603
Media Gateway initializes or selects bearer connection with Far End Media Gateway
604
Bearer Path Established ?
No
Yes
605
Time-out ?
No
Yes
606
Connection not available (cont) event sent to MGC
607
MGC notifies Far End
608
Send transaction reply message to MGC
609
Connection available (coav) signal event to MGC
610
MGC notifies Far End
611
Continuity check (optional)

FIG. 7

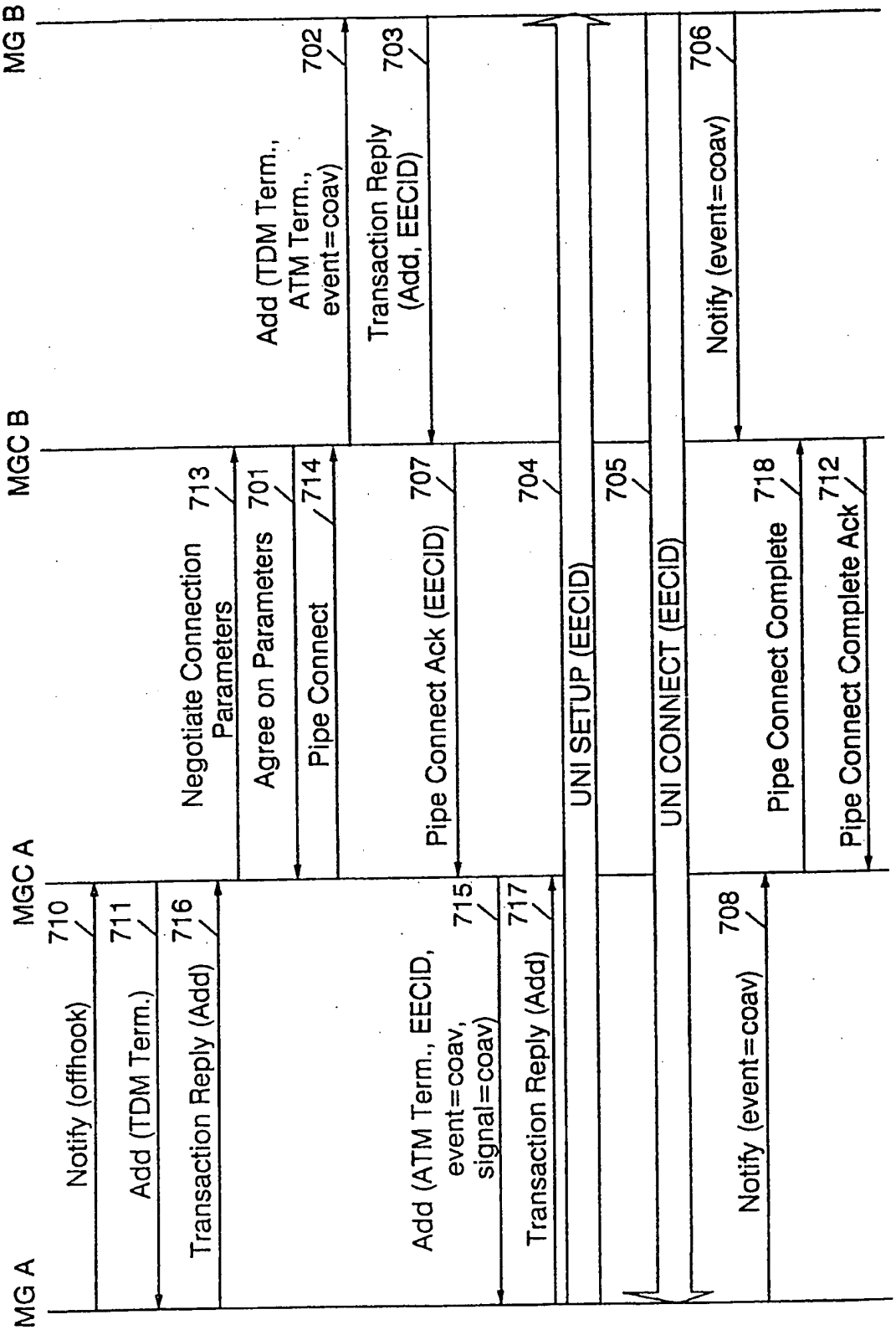

MG A
MGC A
MGC B
MG B
Notify (offhook)
710
Add (TDM Term.)
711
Transaction Reply (Add)
716
Negotiate Connection Parameters
713
Agree on Parameters
701
Pipe Connect
714
Add (TDM Term., ATM Term., event=coav)
702
Transaction Reply (Add, EECID)
703
Add (ATM Term., EECID, event=coav, signal=coav)
715
Pipe Connect Ack (EECID)
707
Transaction Reply (Add)
717
UNI SETUP (EECID)
704
UNI CONNECT (EECID)
705
Notify (event=coav)
708
Pipe Connect Complete
718
Pipe Connect Complete Ack
712
Notify (event=coav)
706

FIG. 8

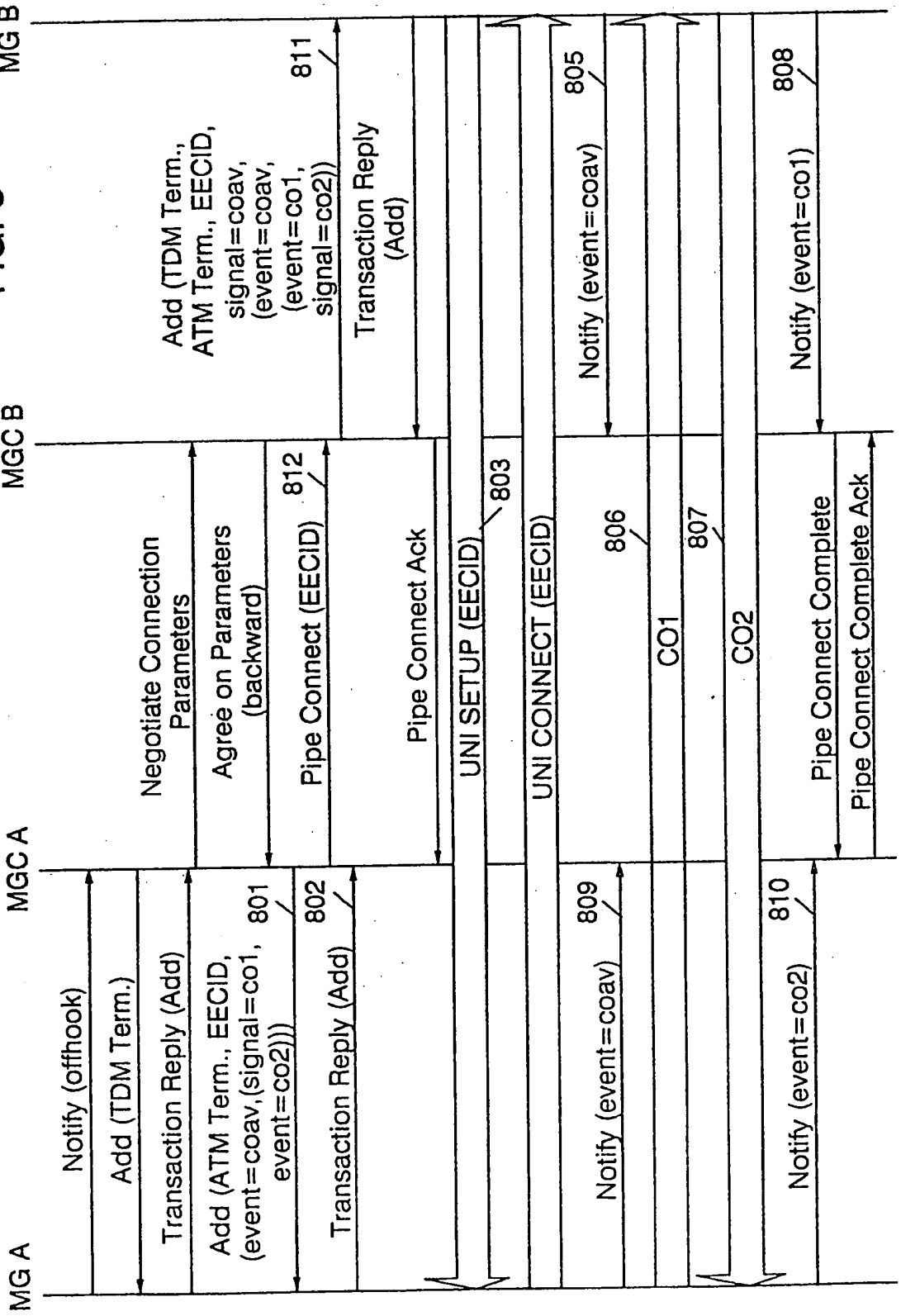

MG A
MGC A
MGC B
MG B
Notify (offhook)
Add (TDM Term.)
Transaction Reply (Add)
Add (ATM Term., EECID, (event=coav,(signal=co1, event=co2)))
801
Transaction Reply (Add)
802
Negotiate Connection Parameters
Agree on Parameters (backward)
Pipe Connect (EECID)
812
Pipe Connect Ack
Add (TDM Term., ATM Term., EECID, signal=coav, (event=coav, (event=co1, signal=co2))
811
Transaction Reply (Add)
UNI SETUP (EECID)
803
UNI CONNECT (EECID)
Notify (event=coav)
809
806
Notify (event=coav)
805
CO1
807
CO2
Notify (event=co2)
810
Notify (event=co1)
808
Pipe Connect Complete
Pipe Connect Complete Ack

FIG. 9

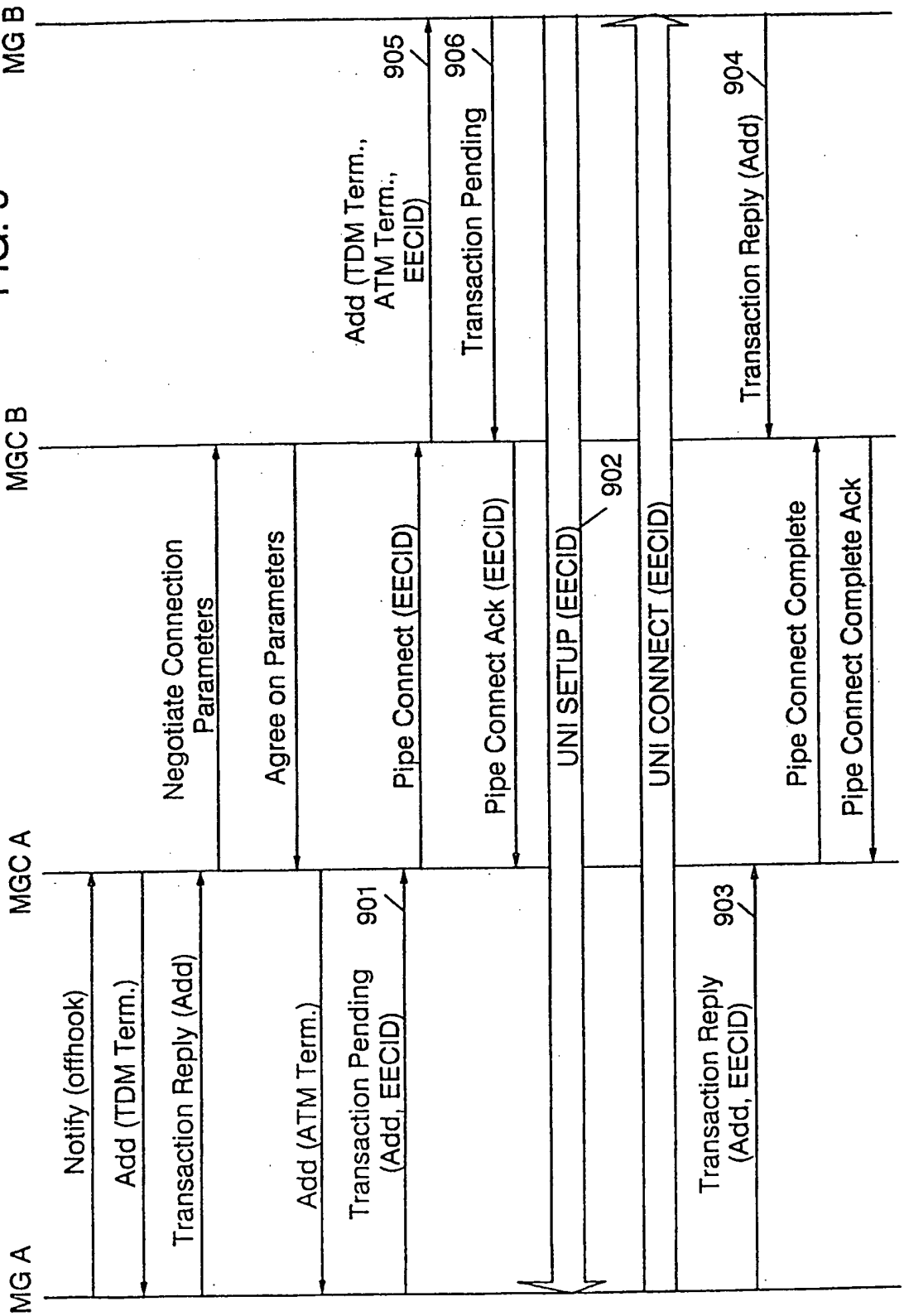

MG A
MGC A
MGC B
MG B
Notify (offhook)
Add (TDM Term.)
Transaction Reply (Add)
Add (ATM Term.)
Transaction Pending (Add, EECID)
901
Negotiate Connection Parameters
Agree on Parameters
Pipe Connect (EECID)
Pipe Connect Ack (EECID)
UNI SETUP (EECID)
902
UNI CONNECT (EECID)
Transaction Reply (Add, EECID)
903
Pipe Connect Complete
Pipe Connect Complete Ack
Add (TDM Term., ATM Term., EECID)
905
Transaction Pending
906
Transaction Reply (Add)
904

FIG. 10

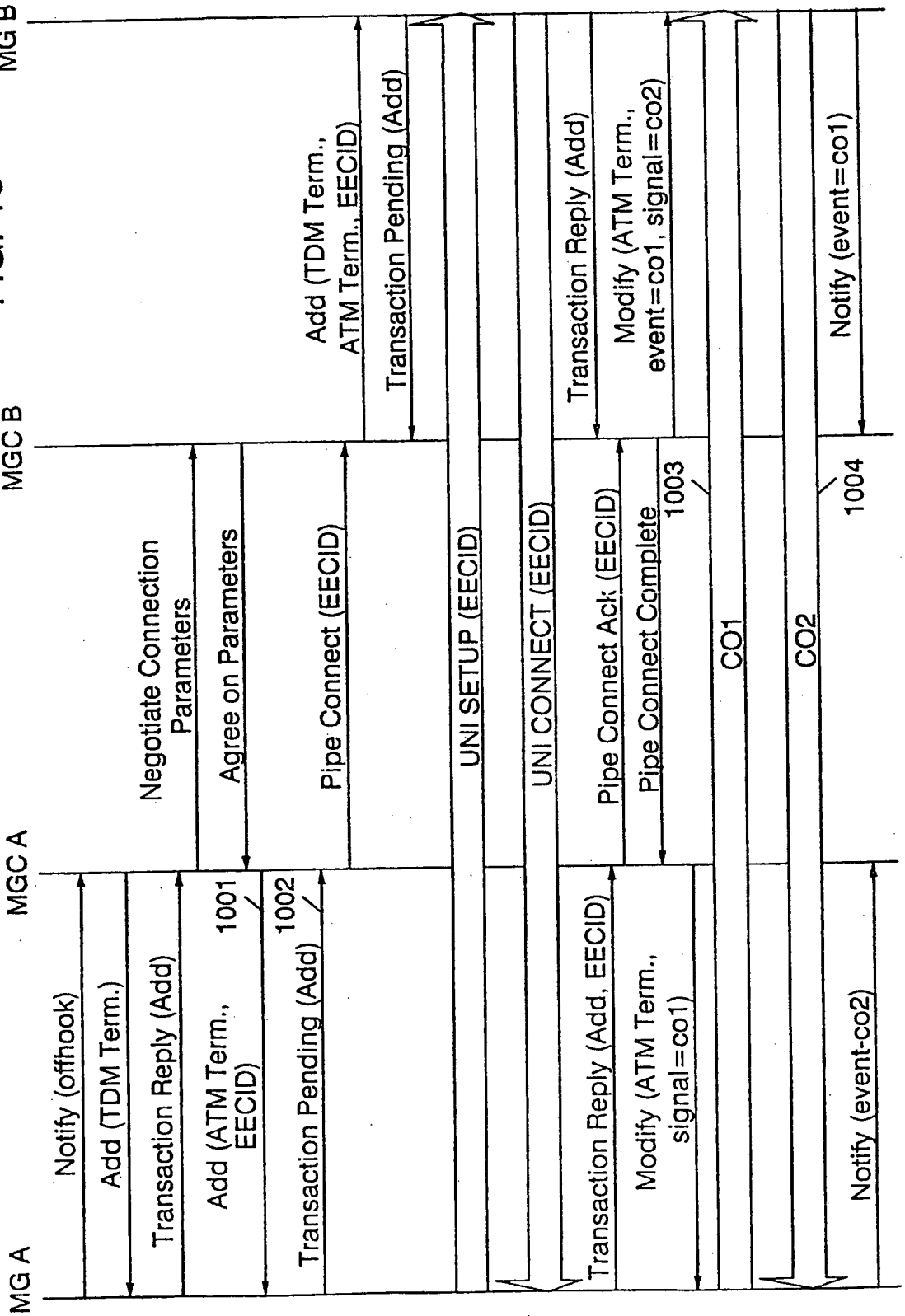

MG A
MGC A
MGC B
MG B
Notify (offhook)
Add (TDM Term.)
Transaction Reply (Add)
Add (ATM Term., EECID)
1001
Transaction Pending (Add)
1002
Negotiate Connection Parameters
Agree on Parameters
Pipe Connect (EECID)
Add (TDM Term., ATM Term., EECID)
Transaction Pending (Add)
UNI SETUP (EECID)
UNI CONNECT (EECID)
Transaction Reply (Add, EECID)
Modify (ATM Term., signal=co1)
Transaction Reply (Add)
Pipe Connect Ack (EECID)
Pipe Connect Complete
1003
Modify (ATM Term., event=co1, signal=co2)
CO1
CO2
1004
Notify (event=co1)
Notify (event-co2)

FIG. 11

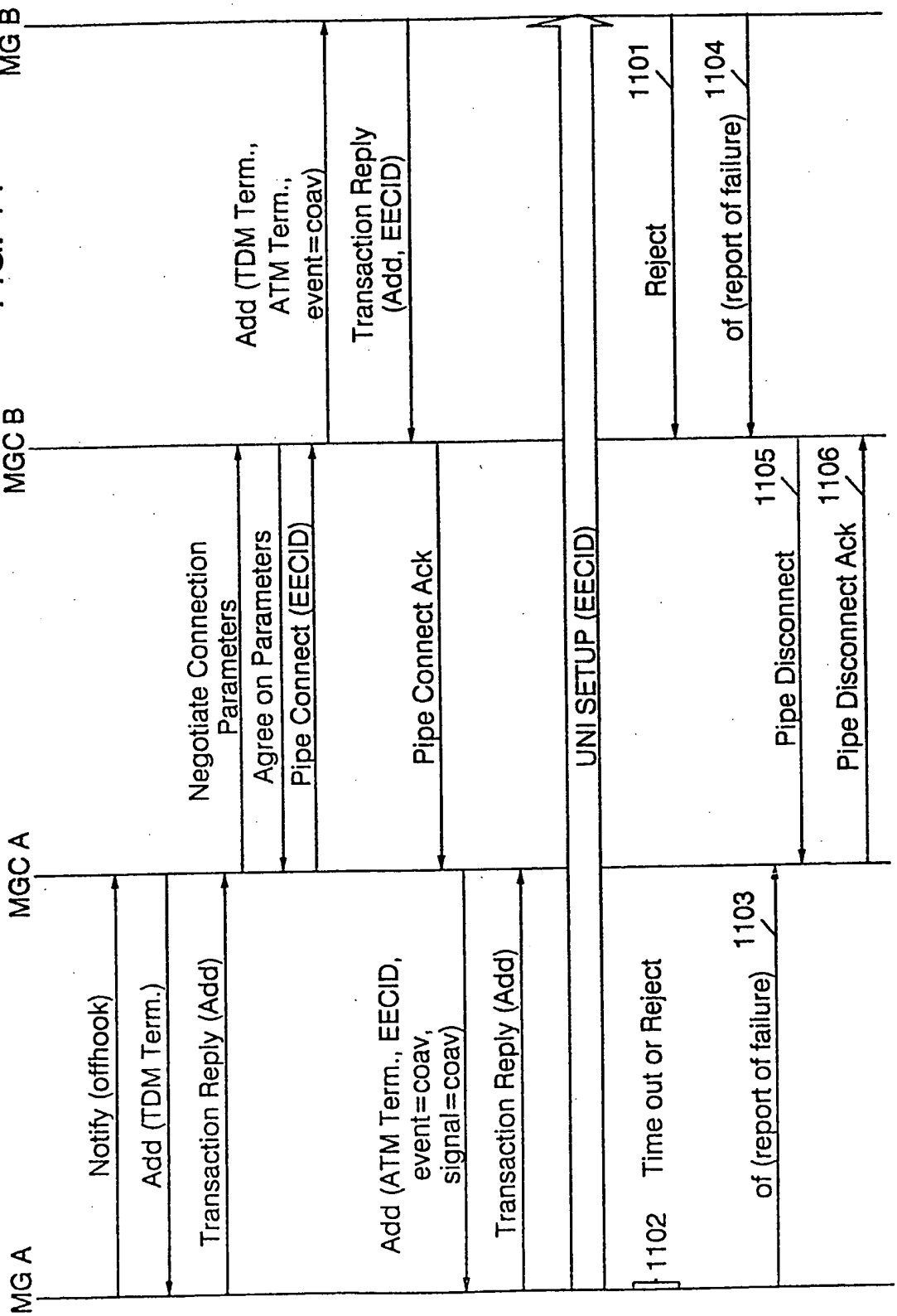
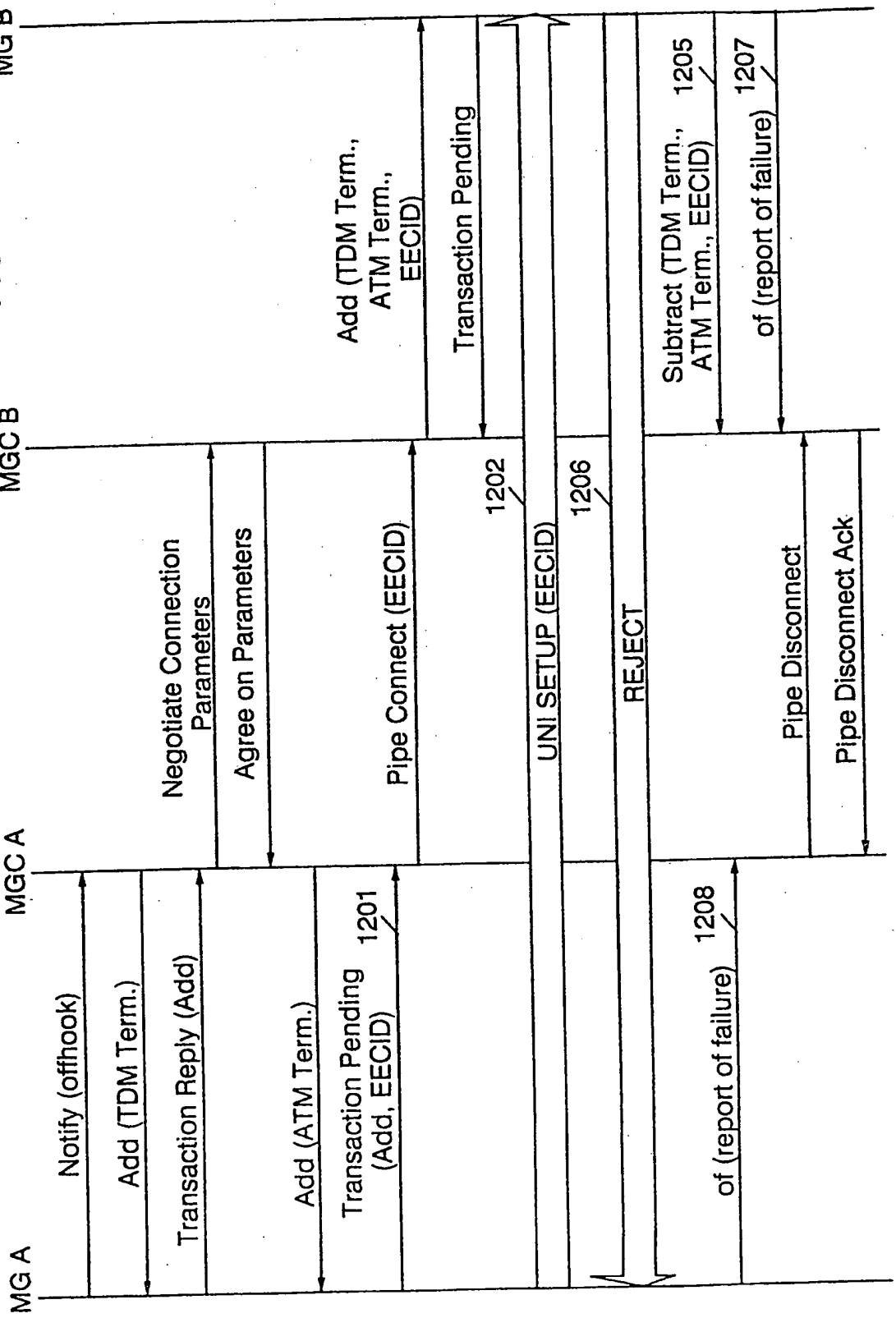

MG A
MGC A
MGC B
MG B
Notify (offhook)
Add (TDM Term.)
Transaction Reply (Add)
Negotiate Connection Parameters
Agree on Parameters
Pipe Connect (EECID)
Add (TDM Term., ATM Term., event=coav)
Transaction Reply (Add, EECID)
Pipe Connect Ack
Add (ATM Term., EECID, event=coav, signal=coav)
Transaction Reply (Add)
UNI SETUP (EECID)
1102 Time out or Reject
1103 of (report of failure)
1101 Reject
1104 of (report of failure)
1105 Pipe Disconnect
1106 Pipe Disconnect Ack

FIG. 12


MG A
MGC A
MGC B
MG B
Notify (offhook)
Add (TDM Term.)
Transaction Reply (Add)
Add (ATM Term.)
Transaction Pending (Add, EECID)
1201
Negotiate Connection Parameters
Agree on Parameters
Pipe Connect (EECID)
Add (TDM Term., ATM Term., EECID)
Transaction Pending
1202
UNI SETUP (EECID)
1206
REJECT
Subtract (TDM Term., ATM Term., EECID)
1205
of (report of failure)
1207
Pipe Disconnect
Pipe Disconnect Ack
1208
of (report of failure)

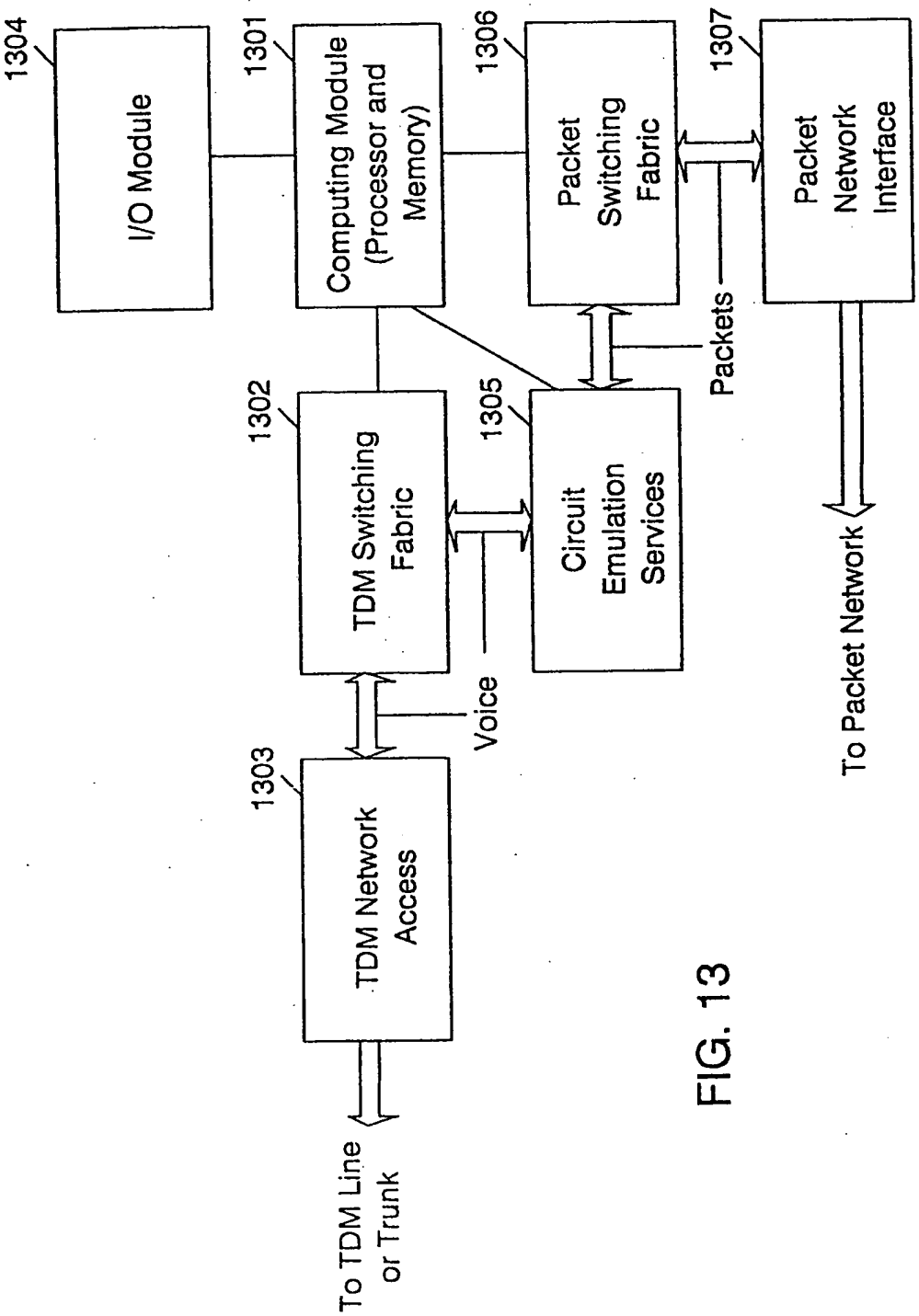
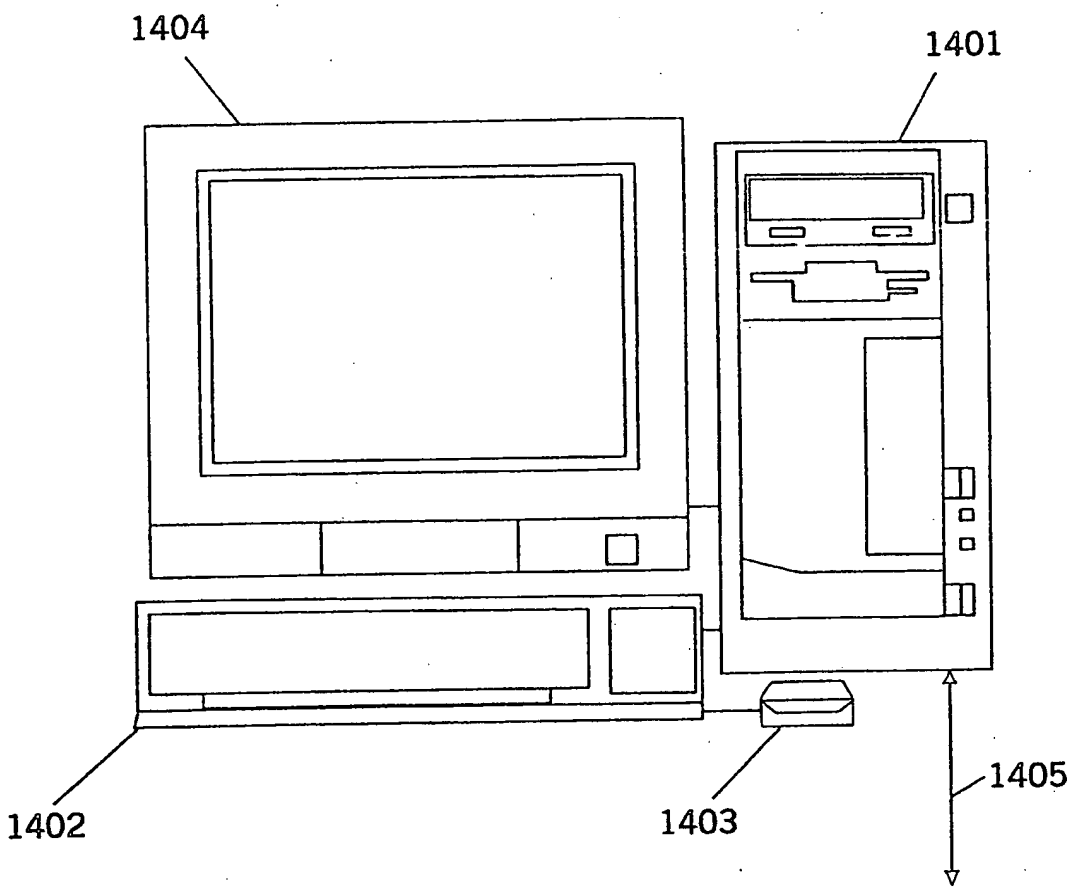

1304
I/O Module
1301
Computing Module (Processor and Memory)
1306
Packet Switching Fabric
1307
Packet Network Interface
1302
TDM Switching Fabric
1305
Circuit Emulation Services
Packets
To Packet Network
Voice
1303
TDM Network Access
To TDM Line or Trunk

FIG. 13

FIG. 14



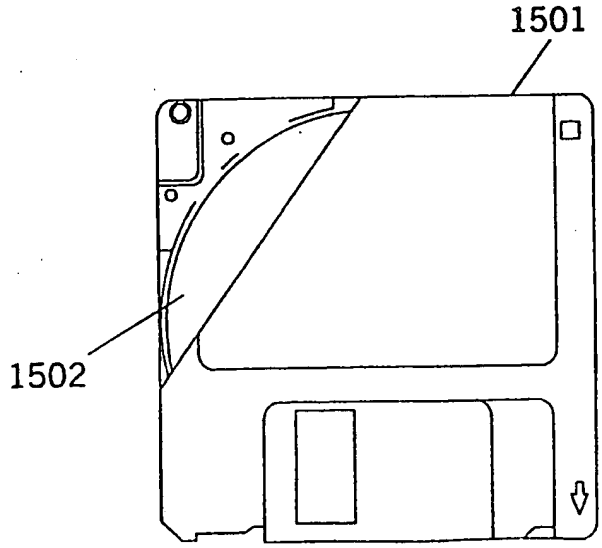


FIG. 15

INTERNATIONAL SEARCH REPORT

Inter nal Application No
PCT/US 00/22853

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04Q11/04 H04L29/06

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04Q H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 682 325 A (GOODMAN WILLIAM ET AL) 28 October 1997 (1997-10-28) abstract column 4, line 31 - line 65 column 12, line 26 - line 39 ---	1,5,7, 11,13, 17, 21-23, 25,29
A	PEARCE ET AL: "CS-2 enhancements on user interaction" EDINBURGH, UK, MARCH 29 - APRIL 1, 1998,LONDON: IEE,UK, vol. NO. 451, 29 March 1998 (1998-03-29), pages 235-239, XP002146298 ISBN: 0-85296-700-4 abstract --- -/--	1-31

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
2 January 2001	12/01/2001

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Staessen, B

1

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

Inter nal Application No
PCT/US 00/22853

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
A	JEFFREY M ET AL: "A SIGNALLING ARCHITECTURE FOR MULTIMEDIA SERVICES" PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON COMMUNICATIONS,US,NEW YORK, IEEE, vol. -, 14 June 1992 (1992-06-14), pages 604-608, XP000326752 ISBN: 0-7803-0599-X abstract	1-31
A	BLANKERS P: "FUNCTIONAL MODELLING FOR TARGET B-ISDN SIGNALLING" PROCEEDINGS OF THE GLOBAL TELECOMMUNICATIONS CONFERENCE (GLOBECOM),US,NEW YORK, IEEE, vol. -, 6 December 1992 (1992-12-06), pages 1166-1171, XP000357735 ISBN: 0-7803-0608-2 abstract	1-31

1

INTERNATIONAL SEARCH REPORT

.ormation on patent family members

Inter nal Application No
PCT/US 00/22853

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5682325 A	28-10-1997	US 5740075 A	14-04-1998
		US 5621728 A	15-04-1997
		US 5748493 A	05-05-1998
		US 5917537 A	29-06-1999